Detecting Vicious Cycles in Urban Problem Knowledge Graph using Inference Rules

Shusaku Egami[1,2†], Takahiro Kawamura[1,3], Kouji Kozaki[1,4] & Akihiko Ohsuga[2]

[1]National Institute of Advanced Industrial Science and Technology (AIST), Tokyo 135-0064, Japan

[2]Graduate School of Informatics and Engineering, The University of Electro-Communications, Tokyo 182-8585, Japan

[3]National Agriculture and Food Research Organization, Ibaraki 305-8517, Japan

[4]Faculty of Information and Communication Engineering, Osaka Electro-Communication University, Osaka 572-8530, Japan

ABSTRACT

Urban areas have many problems, including homelessness, graffiti, and littering. These problems are influenced by various factors and are linked to each other; thus, an understanding of the problem structure is required in order to detect and solve the root problems that generate vicious cycles. Moreover, before implementing action plans to solve these problems, local governments need to estimate cost-effectiveness when the plans are carried out. Therefore, this paper proposed constructing an urban problem knowledge graph that would include urban problems' causality and the related cost information in budget sheets. In addition, this paper proposed a method for detecting vicious cycles of urban problems using SPARQL queries with inference rules from the knowledge graph. Finally, several root problems that led to vicious cycles were detected. Urban-problem experts evaluated the extracted causal relations.

1. INTRODUCTION

Local governments must solve a number of urban problems, including suburban crimes, dead shopping streets, and littering. Thus, local government representatives discuss solutions to these problems. However, because various factors are socially intertwined, urban problems are difficult to solve without understanding

†   Corresponding author: Shusaku Egami (Email: s-egami@aist.go.jp; ORCID: 0000-0002-3821-6507).

the causal relations among them. Thus, structural management of the data on both urban problems and their causality is required for when visualizing and solving such problems. In causal relations, the term "causality" is used to describe the cause-and-effect relationships between two or more factors. To implement action plans, local governments need to grasp the concept of cost-effectiveness. Because most local governments are very cost-sensitive, new projects that seek to solve urban problems cannot be established without clear estimates of their effects (e.g., cost reduction). In fact, this was a comment we received from an official in the Yokohama Policy Bureau, Kanagawa Prefecture, Japan.

Therefore, one of our objectives in this paper is to construct a knowledge graph (KG) that includes the causal relations of urban problems and the related cost information in budget sheets. This KG can predict the impact of urban problems by tracing both causality and the hierarchical links in background ontologies. In addition, in this paper, we aim to detect vicious cycles from among urban problems, to identify these cycles' root problems, and to search the related budget information using the constructed KG. The KG can also help local governments to consider solutions to the intertwined urban problems in terms of their cost effectiveness.

In this study, we first designed an ontology that represents the causality of the urban problems using Web Ontology Language (OWL), and then extended the vocabularies to include budget information based on QB4OLAP [1], which is an extension of the RDF Data Cube vocabulary[①]. Next, we semi-automatically constructed the KG based on the ontology.

For constructing a KG of urban problems, extracting as many words as possible related to the causal relations of urban problems while removing unrelated words is necessary. As a solution to this challenge, (1) we collected various data sources from the Web, including government Web pages, open data, news articles, PDF documents featuring the academic literature, and citizens' voices on blogs and SNS. We then (2) filtered the target to locate sentences containing "factor," "affect," and their synonyms, (3) extracted candidate causal relations by using dependency structure analysis. Finally, we (4) extracted causal word candidates with a certain level of agreement by crowdsourcing using word clouds. In addition, (5) we defined 11 patterns of causal relations that should be considered, and we proposed a method for complementing them using inference rules written in Semantic Web Rule Language (SWRL).

Furthermore, we detected vicious cycles and root problems using SPARQL, then evaluated them based on comments from urban-problem experts. We also detected root problems that lead to multiple vicious cycles and searched the related budget information. However, as there is no absolutely correct dataset related to urban-problem causality, it is difficult to evaluate the detected vicious cycles and the root problems that were estimated using SPARQL queries. Thus, in this paper, we address these evaluations in cooperation with the Osaka City Citizens Bureau and report the results of two case studies. As a result, our contributions are as follows:

---

① https://www.w3.org/TR/vocab-data-cube/

1. Designing an ontology of urban problem causality and local governments' budgets;
2. Extracting urban problem causality from various documents and structuring the data as a KG;
3. Proposing a method for detecting vicious cycles and root problems using SPARQL and SWRL; and
4. Evaluating those vicious cycles and root problems based on comments obtained from urban-problem experts.

The remaining sections of this paper are organized as follows. In Section 2, we provide an overview of knowledge graphs relating to urban problems, city data, and crowdsourcing. In Section 3, we describe the schema design. In Section 4, we outline the method for constructing KGs related to urban problems and evaluate its effectiveness. In Section 5, we describe the detection of vicious cycles and root problems and discuss experts' evaluation in cooperation with the Osaka City Citizens Bureau. Finally, Section 6 concludes this paper with some feasible future extensions.

2. RELATED WORK

2.1 Using Knowledge Graphs to Solve Social Problems

Some studies have proposed the use of linked data to solve social issues. Szekely et al. [2] built knowledge graph from crawled websites and used the data to combat human trafficking and develop a lost children search system that six law enforcement agencies and several NGOs have since deployed [2]. Szekely et al. built knowledge graph related to a specific domain of social problems; however, in this paper, we aim to build a KG related to multiple urban problems including their causality.

In our previous work [3], we built and visualized levels of detail (LOD) to solve the problem of illegally parked bicycles, which is an urgent urban problem in Japan. In addition, we proposed a methodology for designing an LOD schema involving everyday urban problems, such as that one. In this methodology, we completed all the steps manually. In addition, in the previous approach, the number of Web documents that we collected was limited, and the task relied on the workers' knowledge, so it suffered from low coverage with respect to extracting the causality of urban problems. Thus, we propose a method of semi-automatically extracting the causality of urban problems [4]. Moreover, the constructed LOD in the previous study [5] was based on a schema extended from Event Ontology②; however, it was difficult to search for urban-problem causality using OWL inference rules. In addition, the LOD did not contain budgetary information. Thus, in this paper we define an ontology representing urban problem causality and the budget information.

Shiramatsu et al. [6] proposed using LOD to share goals and solve social issues. The resulting goal-matching uses LOD to facilitate civic technology, a field that is aimed at solving social issues using information technology and collaboration between citizens and local governments. Furthermore, this research led to a Web application called GoalShare, which has been used for domestic civic technology

---

② http://motools.sourceforge.net/event/event.html

events. However, Shiramatsu's LOD mainly describes public goals for solving social issues; it does not describe causalities. Associating this LOD with the one proposed in our study will facilitate finding solutions to social problems.

As a result, Szekely et al. [2] and our previous work [3] aimed to solve a specific problem. In contrast, this paper has a cross-domain perspective, which should be common in social problems. Therefore, the schema of our knowledge graphs has been well considered and has the future extendability. Moreover, the objectives of the Shiramatsu et al. [6] are goal-share and goal-matching, and the study differs from ours.

2.2 Using Knowledge Graphs to Analyze City Indicators

Santos et al. [7] defined city knowledge graphs using OWL to analyze various city indicators. They proposed using quality-of-experience ontological indicators, which are calculated numerical values that support convenient visualization. They also developed a dashboard application that can generate widgets to aid in visualizing knowledge graph data. Pileggi et al. [8] defined the ontological framework and implemented it using OWL-DL ontology to represent dynamic, fine-grained urban indicators. To simplify both the understanding of the data structure and the facilitation of its usability, they partitioned the ontology into five sub-ontologies based on the function and scope of the model: indicator, data, profiling, computations and geographic context.

LinkedSpending [9] represents linked data and is based on OpenSpending®, which is an open platform for public financial information, including budgets, spending, balance sheets, and procurement. As of May 2017, users have registered 1,104 datasets from 75 countries, and Japan has the most registered datasets of any country (415). The data is modeled on RDF Data Cube vocabulary, which is designed for modeling multidimensional statistical data. However, the linked data does not describe urban problems and cannot be directly used to solve such problems. Based on the LinkedSpending, we then used QB4OLAP, which is an extension of the RDF Data Cube.

2.3 Crowdsourcing and Natural Language Processing for Linked Data

Demartini et al. [10] proposed an entity linking method using crowdsourcing to improve the quality of links and developed a probabilistic framework to integrate inconsistent results. Celino et al. [11] developed a mobile application to link point of interest data to pictures using crowdsourcing. They also introduced a method of game with a purpose (GWAP) [12] to give users incentives. However, no study has yet included an LOD related to urban-problem causality using crowdsourcing.

Nguyen et al. [13] proposed a method for constructing linked data concerning users' activities. They used conditional random field to extract users' activities from Japanese weblogs and constructed triples of action, object, time, and location. These linked datasets can be applied to analyze users' activities at the

---

® https://openspending.org/

time of an earthquake. LODifier [14] extracted entities from unstructured text using a named entity recognition (NER) system called Wikifier [15], and then combined the entities using DBpedia and WordNet. There are many other studies using natural language processing (NLP) techniques to construct linked datasets. Current state-of-the-art NER systems in English typically have 85% to 90% accuracy for news text such as articles (e.g., CoNLL03 shared task dataset) – but they still perform poorly (about 30%–50% accuracy) on short texts, which do not have implicit linguistic formalism (e.g., punctuation, spelling, spacing, formatting, unorthodox capitalization, emoticons, abbreviations, or hashtags) [16]. Thus this paper combined natural language processing with crowdsourcing to extract urban-problem causality. Although, there are studies that refine linked data using crowdsourcing, those studies differ from ours in combination with NLP.

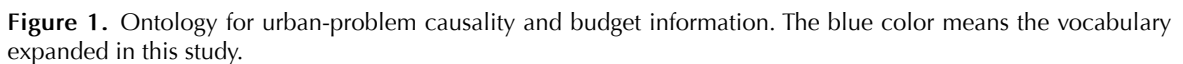### 2.4 Semantic Inference to Detect Relations

In the field of drug-drug interaction (DDI), there are many studies using inference rules to detect new relations in knowledge graph, and we referred their evaluation methods in this paper. Moitra et al. [17] modeled the DDI of pharmacokinetic using Semantic Application Design Language, then estimated interactions related to several enzymes using SWI-Prolog. Herrero-Zazo et al. [18] provided a comprehensive ontology for interactions between pharmacokinetic and pharmacodynamic (DINTO), then estimated the relations such as "may interact with" using the DINTO and SWRL rules. Many other studies are related to ontological reasoning; however, to the best of our knowledge, no studies have used inference rules to infer the causal relations among urban problems.

3. DESIGNING AN ONTOLOGY OF PROBLEM CAUSALITY AND COSTS

Our KG is mainly meant to be used in the investigation of solutions to urban problems.

Specifically, local governments can query our KG to consider these solutions' effects and budgets. Thus, we designed the ontology shown in Figure 1 to represent urban problem causality and local governments' budgets.

The upper half of Figure 1 defines the vocabulary representing urban-problem causality. In this part, all resources are classified as `upv:CausalEntity` and a subset of them as `upv:UrbanProblem`. There are two main causality properties, `upv:factor` and `upv:affect`. Both are subproperties of the `upv:related` property. Because urban problems are not events and thus do not have temporal or spatial aspects, we did not reuse the `event:factor` property in the Event Ontology. The sub-properties of `upv:factor` and `upv:affect` represent crowdsourcing agreements; dividing the causality properties into `upv:factor` and `upv:affect` enables forward or backward chainings with agreement levels that restrict the domain or range. For example, when users extract strong causality, the `upv:factor_level4` and `upv:affect_level4` properties can be used. The values of `upv:affect_level4` are words that more than 35 crowdsourcing workers selected as factors influencing the urban problem. When users extract causality (regardless of their agreement), the `upv:factor` and `upv:affect` properties can be used in SPARQL queries.

Figure 1. Ontology for urban-problem causality and budget information. The blue color means the vocabulary expanded in this study.

The lower half of Figure 1 defines the vocabulary that represents budget information. Because most local governments' budget information is published as tabular data (in formats such as Microsoft Excel and PDF), we described it using QB4OLAP, which is an extension of the RDF Data Cube vocabulary. The QB4OLAP has been used in the data models of business-intelligence tools and includes `qb4o:LevelProperty` and `qb4o:AggregateFunction`, which support aggregation operations. Hence, users can query the total budget of each department and determine which urban problem requires the highest budget. The `upq:Project` class means the project for solving social problems, and the instances have at least one `dcterms:subject` property. The values of `dcterms:subject` are instances of the `upv:CausalEntity` class. The instances of the Observation class are cells (values) in tabular data of projects for solving social problems. Therefore, the instances of the Observation class have a `upq:budget` property, and the value of it is the budget. A ward that allocated a project's budget is described as an instance of the `upq:Ward` class,

and a city is described as an instance of `upq:City`. The `rdfs:range` of the `upq:ward` is the `upq:Ward`. Lower-case letters are properties. By designing the schema this way, users can query projects' budgets using aggregate functions.

4. BUILDING A KG USING THE DESIGNED ONTOLOGY

4.1 Extraction of Urban Problem Causality

In this section we propose a method for semi-automatically extracting the causality of urban problems, as follows:

(1) Collect Web documents using a search engine;
(2) Extract causality words from the collected documents using natural language processing;
(3) Generate word clouds based on the extracted words; and
(4) Filter the extracted words using crowdsourcing.

Our goal is to aggregate the qualitative causal knowledge of urban problems from various data sources, such as government Web pages, open data, news articles, the academic literature, blogs, and social networking sites. We also believe that we need to consider human subjective choices. We envision that the main applications will include discussion tools among experts for solving urban problems, as well as tools for explaining the evidence of causal relationships. Thus, the constructed causal relations must exist within a common understanding to some extent, be explainable, and also contain some unexpected results. Therefore, we believe that crowdsourcing using word clouds, which presents cause-and-effect relationship candidates, is suitable for extracting meaningful causality words.

We collected documents from a search engine using the names of urban problems and synonyms of the word "factor" as keywords. For example, the first keyword is "suburban crime," and the second keyword is "factor" (along with its synonyms, which include "element", "origin", and "cause"). We obtained the synonyms of the second keyword from Japanese WordNet and obtained the document lists using Google Custom Search API and Bing Web Search API. We separately collected 50 HTML files and 50 PDF files for each keyword set (i.e., unique combination of the first and second keywords). We collected HTML and PDF files separately and included reports from both governments and citizens. However, we also collected many unrelated documents in this step; thus, we excluded the documents that contained few words related to the urban problems' names. The number of documents that included the urban problem words was specifically as follows: ((kinds of urban problems × # of synonyms of "factor" × 50 HTMLs) + (kinds of urban problems × # of synonyms of "factor" × 50 PDFs) + (kinds of urban problems × # of synonyms of "affect" × 50 HTMLs) + (kinds of urban problems × # of synonyms of "affect" × 50 PDFs)) – # of noise documents = 3,903.

Next, we extracted noun words using morphological analysis. To facilitate the subsequent crowdsourcing process, we concatenated the verbal nouns and constructed noun phrases. For example, the phrase "preventing delinquency" was split into "preventing" and "delinquency" using morphological analysis, but we concatenated these words to a noun phrase in this study.

Then, using Japanese dependency analysis we extracted noun phrases that had causal relationships with the synonyms of the word "factor" based on dependency relations in each sentence [19].

Likewise, we extracted affecting words of urban problems, using the synonyms of "affect" such as "influence", "effect", and "evoke" as the second keywords. We generated word clouds based on the extracted possible causality words and filtered those words using crowdsourcing. We assumed in this step that the word clouds would increase the impression that the words made, thus simplifying the extraction of the important words. In fact, we received comments that this method was fun and game like.

If a word is counted as a different word due to spelling inconsistencies, it will reduce the word cloud's visibility, which needs to be avoided. We used Jaro-Winkler distance [20] to calculate the words' similarity, and empirically set the threshold to 0.8. When we found similar words, we integrated the number of occurrences of those words to the longest word. Figure 2 shows a word cloud of suburban crime factors.
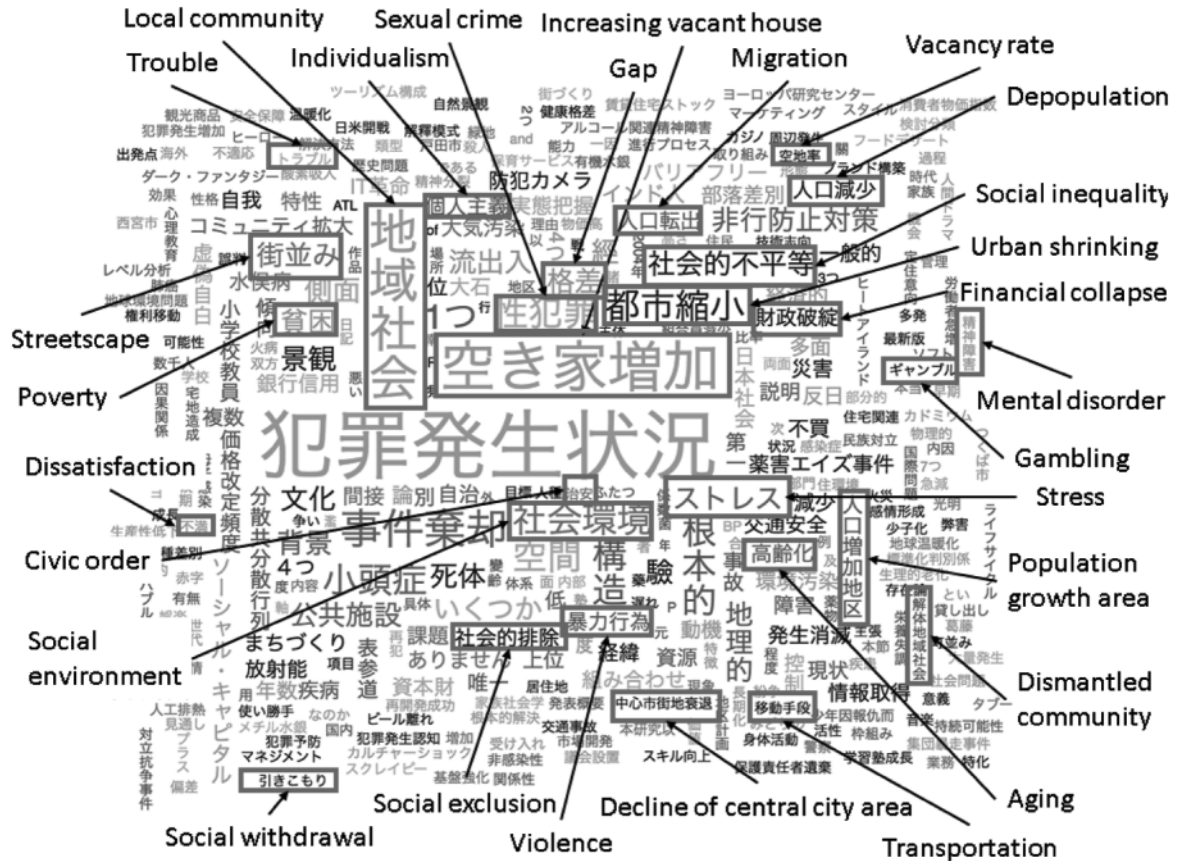
Figure 2. A word cloud of suburban crime factors.

Words with higher frequency are larger and placed closer to the center of the cloud. The color of the words is random. Then, we conducted two crowdsourcing tasks: "select 10 words that are considered factors in suburban crime" and "select 10 words that are considered to be affected by suburban crime." In this paper, we used the Lancers® crowdsourcing service. We set the reward for the two tasks at 50 JPY, and asked up to 50 people to work on each problem. Then, we gathered a list of all words that more than 10% of the workers had selected. Those words are translated in Figure 2.

Furthermore, to enrich the causality of the urban problems, we repeated our method using the extracted causality words. The repetition of this method increased the intermediate nodes in our knowledge graph. However, not all the words had causal relations. Thus, we also extracted cooccurring words from the top 50 Web documents related to the causality words. If we found more than 336 cooccurring words (top 5%), such as "cause", "factor", "influence", "urban", and "city", we applied our extraction method to the causality words.

4.2 Building KG Based on the Extracted Causality Words

We built the KG based on the designed ontology and used the extracted words. Because Lancers exports its results in CSV format, we used Apache Jena® to convert the CSV file to an RDF file based on the designed schema. Specifically, we created urban-problem resources as sub-classes of the `upv:UrbanProblem` class and created other resources as sub-classes of the `upv:CausalEntity` class. In this study, we used both SKOS and OWL due to the usability. Naive users, who do not know much about ontology, can intuitively search the graph by tracing simple SKOS relations (Boarder and Narrower). This does not violate the QB4OLAP's restriction related to SKOS. Needless to say, OWL reasoning is useful for detecting vicious cycles and root problems. We formed the causality links using the `upv:factor` and `upv:affect` properties that corresponded to the number of participants who agreed.

In addition, if resources had the same name as an extracted noun word or matched its values of `skos:altLabel` in WikiData®, we also created alternative resources for extracted noun words, as well as alternative hyper resources. For example, the "temporary work" resource® in WikiData has Japanese labels such as "パートタイマー" (parttime work); it also has a hyper class called "employment"®, which has Japanese labels such as "雇用契約" (employment contract). Figure 3 shows the generated KG fragments. If we found no resources that had the same name as a causal entity or that matched its `skos:altLabel` value in WikiData, we extracted noun words from the name of each class using morphological analysis and then created hyper classes based on those noun words. We used the head and modifier matching methods [21].

---

④ https://www.lancers.jp

⑤ https://jena.apache.org

⑥ https://www.wikidata.org/wiki/Wikidata:Main_Page

⑦ http://www.wikidata.org/entity/Q667944

⑧ http://www.wikidata.org/entity/Q656365

```
PREFIX upr: <http://www.ohsuga.lab.uec.ac.jp/urbanproblem/resource/>
PREFIX upv: <http://www.ohsuga.lab.uec.ac.jp/urbanproblem/vocabulary#>
PREFIX prov: <http://www.w3.org/ns/prov#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
upr:Temporal_employment    a        owl:Class ;
        rdfs:label        "Temporal employment"@en , "非正規雇用"@ja ;
        rdfs:seeAlso     <http://www.wikidata.org/entity/Q667944> ;
        rdfs:subClassOf       upr:Employment , upv:CausalEntity ;
        skos:broader  upv:CausalEntity, upr:Employment ;
        upv:factor_level1       upr:Homeless ;
        prov:alternateOf        upr:Part_timer .
upr:Part_timer          a        owl:Class ;
        rdfs:label        "Part-timer"@en , "パートタイマー"@ja ;
        rdfs:subClassOf       upv:CausalEntity ;
        skos:broader  upv:CausalEntity ;
        prov:alternateOf        upr:Temporal_employment .
upr:Employment          a        owl:Class ;
        rdfs:label        "Employment"@en , "雇用"@ja ;
        rdfs:subClassOf       upv:CausalEntity ;
        skos:broader  upv:CausalEntity ;
        prov:alternateOf        upr:Employment_contract .
upr:Employment_contract     a        owl:Class ;
        rdfs:label        "Employment contract"@en , "雇用契約"@ja ;
        rdfs:subClassOf       upv:CausalEntity ;
        skos:broader  upv:CausalEntity ;
```

Figure 3. Generated KG fragment.

4.3 Generating Instances Based on Budget Data of Local Government

Osaka is an ordinance-designed city in Japan, and it is the capital city of Osaka Prefecture. Osaka City has published various open datasets on its Osaka City Open Data Portal Site®. Most of this site's open datasets that contain budget information are CC-BY 4.0 licensed. First, we used Apache Jena and Apache POI® to convert this site's tabular data and PDF files to RDF files (based on the designed schema).

Next, we linked the local government project resources to the causality resources. Because we had no detailed descriptions of the projects in the source budget sheet, we had to link the project resources to the causality resources using the projects' name. However, we had difficulty determining relations such as the one between a "project for solving the problem of street smoking" and the one focused on "cigarettes". Thus, we linked the project resources to the causality resources using Algorithm 1.

® https://data.city.osaka.lg.jp/

® https://poi.apache.org/

Algorithm 1. Linking the business resources to the causality resources.

Require: $businessList, causalityList, stopwordList$
Ensure: RDF
1: **for** each $business \in businessList$ **do**
2:      $morphemeList_{business} \Leftarrow morphologicalAnalysis(business.label)$
3:      **for** each $morpheme_{business} \in morphemeList_{business}$ **do**
4:          **if** $morpheme_{business}.partOfSpeech == Noun$
              $\&\& !stopwordList.contains(morpheme_{business}.text)$ **then**
5:              **if** $causalityList.contains(morpheme_{business}.text)$ **then**
6:                  $causality \Leftarrow causalityList.get(morpheme_{business}.text)$
7:                  $RDF.addStatement(business, dct : subject, causality)$
8:              **end if**
9:              //Obtain synonym words of the noun words in the business name
10:             $synonymList \Leftarrow getSynonymList(morpheme_{business}.text)$
11:             **for** each $synonym \in synonymList$ **do**
12:                 **if** $causalityList.contains(synonym)$ **then**
13:                     $causality \Leftarrow casaulityList.get(synonym)$
14:                     $RDF.addStatement(business, dct : subject, causality)$
15:                 **end if**
16:             **end for**
17:             //Obtain short sentences describing the word sense
18:             $gloss \Leftarrow getGloss(morpheme.text)$
19:             $morphemeList_{gloss} \Leftarrow morphologicalAnalysis(gloss)$
20:             **for** each $morpheme_{gloss} \in morphemeList_{gloss}$ **do**
21:                 **if** $morpheme_{gloss}.partOfSpeech == Noun$
                      $\&\& !stopwordList.contains(morpheme_{gloss}.text)$ **then**
22:                     **if** $causalityList.contains(morpheme_{gloss}.text)$ **then**
23:                         $causality \Leftarrow casaulityList.get(morpheme_{gloss}.text)$
24:                         $RDF.addStatement(business, dct : subject, causality)$
25:                     **end if**
26:                 **end if**
27:             **end for**
28:         **end if**
29:     **end for**
30: **end for**

In this algorithm, we first extracted all noun words except for stop words from the names of the local projects. Then, we obtained synonyms that corresponded to the extracted noun words using Japanese WordNet; we used these synonyms as the linking candidates. In addition, we obtained glosses of the noun words. A gloss consists of multiple short sentences that describe the word's senses and use of the word. Thus, from these short sentences, we also extracted any noun words that were linking candidates. If the candidate words matched the causality resources, we linked the project resources to the causality resources using the `dctetms:subject` property.

Figure 4 shows part of the KG that we constructed in this study. The resulting KG is accessible from our website⑪. In addition, the source code for collecting the data and building the KG is now available on GitHub⑫. There are 70,076 triples in the ontology. We validated our KG using RDFUnit [22], which is a test-driven data-debugging framework. We used this framework to automatically generate 68 test cases, all of which passed. There were no timeouts, errors, or violations. Therefore, we correctly reused the existing vocabulary without violating the domain or range restrictions. All the resources are linked, and none are independent.

Figure 4. Part of constructed ontology.

4.4 Result of NLP

Table 1 shows the statistics for the extraction of the causality words. Because there were many synonyms of the word "affect," the number of documents related to affecting words was 2,465, which is larger than the number of the documents related to factors (1,438). We excluded some synonyms of "factor" (e.g., "procatarxis") because they are rarely used, which led to search results that contained many unrelated documents. As a result, the number of affecting words was large, and the agreement between the selections was lower than that for the factor words.

---

⑪ http://www.ohsuga.lab.uec.ac.jp/urbanproblem/
⑫ https://github.com/Ease112/urban_problem_kg

Table 1. Statistics for the causality-words extraction.

	# of documents including urban problem words	# of sentences including synonyms of "factor" and "affect"	# of extracted words
Factor	1,438	4,481	3,110
Affect	2,465	9,082	4,661

Missing words related to urban problems can lead to lower agreement in the process of causality-word extraction. In some cases, a phrase extracted using our method did not match the phrase that described the causality. Because there are many complex sentences in government documents, we could not extract the causality words in many cases. To solve this problem, we tried several methods of text simplification. In other cases, the causality words extracted from the descriptions were not related to urban problems. To exclude these errors, we extracted words that appeared in multiple documents instead of those that appeared many times in a single document.

In this study, Jaro-Winkler is used only to solve the spelling inconsistencies displayed in the word cloud. In addition, we used WikiData to obtain the `skos:altLabel` value to take synonyms into account (Section 4.2). On the other hand, we also need to consider the unification of phrases with low string similarity but the same meaning. For example, the use of word embedding techniques may solve this problem. It is possible to calculate the similarity between them after obtaining the vector representation of the causal word candidates. However, since the dependency structure analysis extracts the noun phrases, we need to obtain a vector of noun phrases. Therefore, we need to generate embedding models of noun phrases based on our collected data instead of pre-trained word embedding models.

4.5 Crowdsourcing Results

To calculate the agreement of the causality-word selection through crowdsourcing, we used Fleiss's kappa [23]. We set the number of users to 50; the average number of extracted factor words and affecting words was 0.291 and 0.212, respectively. The total average agreement was 0.256, which indicated fair agreement according to the benchmark [24].

The high agreement for the traffic accident factor (0.443) was due to the various instances of traffic accidents that the Metropolitan Police Department, educational institutions, and news organizations reported this resulted in the workers having extensive background knowledge of the issue. On the other hand, the high agreement for the noise factor (0.468) is because the workers had their own experiences in which noise affected them.

5. DETECTING VICIOUS CYCLES OF URBAN PROBLEMS

5.1 Complementing Missing Links Using Causal Inference Rules

In this paper, we aim to detect vicious cycles of urban problems. However, as our KG included many missing causal links, we considered most vicious cycles as being undetectable from direct causal links alone. Thus, we defined causal inference rules that complemented the missing links. Figure 5 shows the complementary missing links based on hyper classes and alternative classes; the numbers in the figure correspond to inference rules that were described in the SWRL rules below, which were stored to Stardog®, an RDF database that supports OWL and rule reasoning. Because Stardog recommends using native Stardog rules syntax (which is based on SPARQL rather than SWRL), we converted these SWRL rules to Stardog rules as shown below:

```
[1]  (?x upv:affect ?y), (?y skos:broader ?yp) -> (?x upv:probablyAffect ?yp).
[2]  (?x upv:affect ?y), (?x skos:broader ?xp) -> (?xp upv:probablyAffect ?y).
[3]  (?x upv:affect ?y), (?x skos:broader ?xp), (?y skos:broader ?yp) -> (?xp upv:
     mayAffect ?yp).
[4]  (?x upv:affect ?y), (?y prov:alternateOf ?yalt) -> (?x upv:likelyAffect
     ?yalt).
[5]  (?x upv:affect ?y), (?y skos:broader ?yp), (?yp prov:alternateOf ?ypalt) ->
     (?x upv: mightAffect ?ypalt).
[6]  (?x upv:affect ?y), (?x skos:broader ?xp), (?y prov:alternateOf ?yalt) ->
     (?xp upv:mightAffect ?yalt).
[7]  (?x upv:affect ?y), (?x skos:broader ?xp), (?y skos:broader ?yp), (?yp prov:
     alternateOf ?ypalt) -> (?xp prov:possiblyAffect ?ypalt).
[8]  (?z upv:affect ?yalt), (?yalt prov:alternateOf ?y), (?y skos:broader ?yp) ->
     (?z upv:mightAffect ?yp).
[9]  (?z upv:affect ?yalt), (?yalt prov:alternateOf ?y), (?y skos:broader ?yp),
     (?yp prov:alternateOf ?ypalt) -> (?z upv:possiblyAffect ?ypalt).
[10] (?z upv:affect ?yalt), (?z skos:broader ?zp), (?yalt prov:alternateOf ?y),
     (?y skos:broader ?yp) -> (?zp upv:possiblyAffect ?yp).
[11] (?z upv:affect ?yalt), (?z skos:broader ?zp), (?yalt prov:alternateOf ?y),
     (?y skos:broader ?yp), (?yp prov:alternateOf ?ypalt) -> (?zp upv:possiblyAffect
     ?ypalt).
```
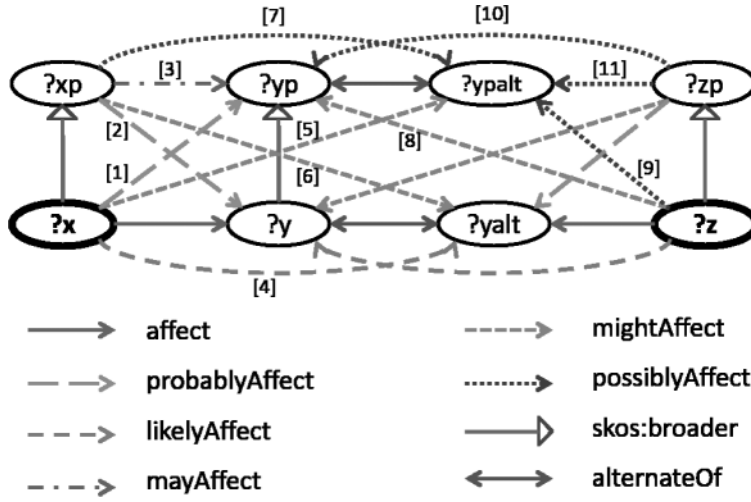

® https://www.stardog.com/

Figure 5. Inference properties for complementing missing causal relations.

These rules created five causal relation properties: probablyAffect, likelyAffect, mayAffect, mightAffect, and possiblyAffect.

We defined the cost of "upv:affect" as 1, the cost of "prov:alternateOf" as 0.75, and the cost of "skos:broader" as 0.5. Therefore, we defined the strength of the causality for the inference properties as the total costs of the antecedent properties such that probablyAffect > likelyAffect > mayAffect > mightAffect > possiblyAffect. These properties are subproperties of "upv:affect."

As a result of the experiment, we complemented 1,058 probablyAffect properties, 122 likelyAffect properties, 191 mayAffect properties, 333 mightAffect properties, and 179 possiblyAffect properties.

5.2 Detecting Vicious Cycles of Urban Problems

We defined the vicious cycle of urban problems as a loop of three or more nodes using only sub-properties of the `upv:affect` (Figure 6). Each node corresponds to a subclass of either `upv:UrbanProblem` or `upv:CausalEntity`. At least one of these nodes is an urban problem. To detect the vicious cycles of urban problems, we used SPARQL queries to extract the cycles that contained 3 to 6 nodes. The limit of maximum cycle length can be incrementally increased in consultation with experts in practice. Figure 7 shows an example SPARQL query for detecting 3-node vicious cycles. Moreover, as the obtained vicious cycles included duplicates such as "Poverty → Truancy → Disease" and "Truancy → Disease → Poverty," we deleted such duplicates.
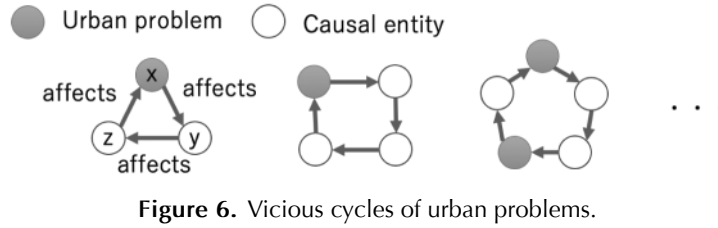
Figure 6. Vicious cycles of urban problems.

Figure 7. An example SPARQL query for detecting 3 nodes vicious cycles.

Table 2 shows the number of detected vicious cycles; we detected 951 vicious cycles through SPARQL queries and 1,904 vicious cycles through SPARQL queries with inference rules. The "Inference" column shows the results of the SPARQL queries after we applied the inference rules described in Section 5.1. When we search vicious cycles and root problems using SPARQL, we changed the type of upv:affect from owl:TransitiveProperty to owl:ObjectProperty. Thus, the arbitrary long cycles do not appear in the results of 3 nodes. Also, the results of 3 nodes (duplicates) are surely removed from the results of others. Therefore, our ontology is based on OWL DL, and the reasoning is sound.

Table 2. The number of detected vicious cycles.

Vicious cycles	No influence	Inference
3 nodes	33	45
4 nodes	168	308
5 nodes	236	460
6 nodes	514	1,091
Total	951	1,904

5.3 Experiment for Detecting Root Problems Using SPARQL Patterns

Next, we used SPARQL queries to detect the root problems that led to multiple vicious cycles. Figure 8 shows the query for detecting these root problems that affected two vicious cycles. Figure 9 shows the query for detecting root problems included in two vicious cycles. Figure 10 shows the graph patterns. The left side is the root problem obtained from the query in Figure 8. The right side is the root problem obtained from the query in Figure 9. The number of viciouscycle nodes was set to between 3 and 6. As a result, we obtained 144 graph patterns of root problems and detected 28 root problems.

```
PREFIX upv: <http://www.ohsuga.lab.uec.ac.jp/urbanproblem/vocabulary#>
SELECT DISTINCT ?x ?y ?z ?a ?b ?c WHERE {
        ?x      upv:affect      ?y .
        ?y      upv:affect      ?z .
        ?z      upv:affect      ?x .
        ?a      upv:affect      ?b .
        ?b      upv:affect      ?c .
        ?c      upv:affect      ?a .
        ?root   upv:affect      ?x ;     upv:affect      ?a .
        ?y      rdfs:subClassOf         ?yp .
        ?b      rdfs:subClassOf         ?bp .
        filter(?yp = upv:UrbanProblem)
        filter(?x != ?y && ?x != ?z && ?y != ?z)
        filter(?bp = upv:UrbanProblem)
        filter(?x != ?b && ?x != ?c && ?b != ?c && ?y != ?b && ?z != ?c)
}
```

Figure 8. SPARQL query for detecting root problem; one root problem affects two vicious cycles.

```
SELECT DISTINCT ?x ?y ?z ?b ?c WHERE {
        ?x      upv:affect      ?y .
        ?y      upv:affect      ?z .
        ?z      upv:affect      ?x .
        ?x      upv:affect      ?b .
        ?b      upv:affect      ?c .
        ?c      upv:affect      ?x .
        ?y      rdfs:subClassOf         ?yp .
        ?b      rdfs:subClassOf         ?bp .
        filter(?yp = upv:UrbanProblem)
        filter(?x != ?y && ?x != ?z && ?y != ?z)
        filter(?bp = upv:UrbanProblem)
        filter(?x != ?b && ?x != ?c && ?b != ?c && ?y != ?b && ?z != ?c)
}
```

Figure 9. SPARQL query for detecting root problem; two vicious cycles share a root problem.
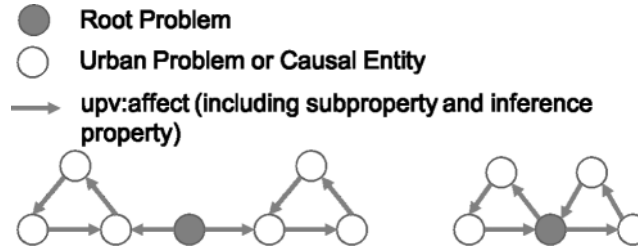
Figure 10. Graph patterns of root problems.

5.4 Evaluation of the Detected Vicious Cycles

In our previous study [26], we defined vicious cycles as only consisting of correct causal relations and assumed that the relations described in official government documents were correct. However, governments can publish rather limited information, so the dataset of correct relations was incomplete. Therefore, for this study, we evaluated the causal relations in cooperation with experts who were working on solving various urban problems; this included experts on homelessness and crime prevention as well as representatives of the Osaka Citizens Bureau. We then evaluated the results related to homelessness and crime from our questionnaires and interviews. Figure 11 depicts the interview conducted at Osaka City Citizens Bureau on January 25, 2018. The six experts are affiliated with NPOs, companies, the Institute for Municipal Research, and the Osaka City Citizens Bureau. Specifically, the experts gave one of the following four options on 194 causal relations related to homelessness and crime:

(1) The extracted causal relation is true.
(2) The extracted causal relation might be true (including new knowledge).
(3) The extracted causal relation is false.
(4) Additional causal relations were not extracted but should be added.

The experts chose Option (1) 21 times, Option (2) 154 times, Option (3) 5 times, and Option (4) 14 times. The answers were published on our website®. For example, the experts classified the extracted triple "Multiple debt $\xrightarrow{\text{affects}}$ Homeless" Option 1 and the extracted triple "Homelessness $\xrightarrow{\text{affects}}$ Environmental pollution" as Option 2, based on the idea that homeless people tend to scatter plastic trash when collecting scraps. The experts classified the extracted triple "Population aging $\xrightarrow{\text{affects}}$ Homelessness" as Option 3 because the aging of homeless people is a problem but is not a factor in homelessness. As an example of Option 4, one expert commented that the nuclear family is a factor in crimes, but we could not extract the term "nuclear family" as a factor in crimes. The meaning of Option (2) "might be true", which means "Experts knew it, but could not affirm it with confidence" or "Experts did not know it but can consider it new possibilities". Therefore, the selection of this option indicates that the causal reasoning offered new insights to the experts. Then, the experts can consider the problems based on the hypothesis obtained from causal relations. The purpose of our study is to suggest such a hypothesis to experts. Thus the evaluation can be interpreted as a success.

® http://www.ohsuga.lab.uec.ac.jp/urbanproblem/evaluation2.html
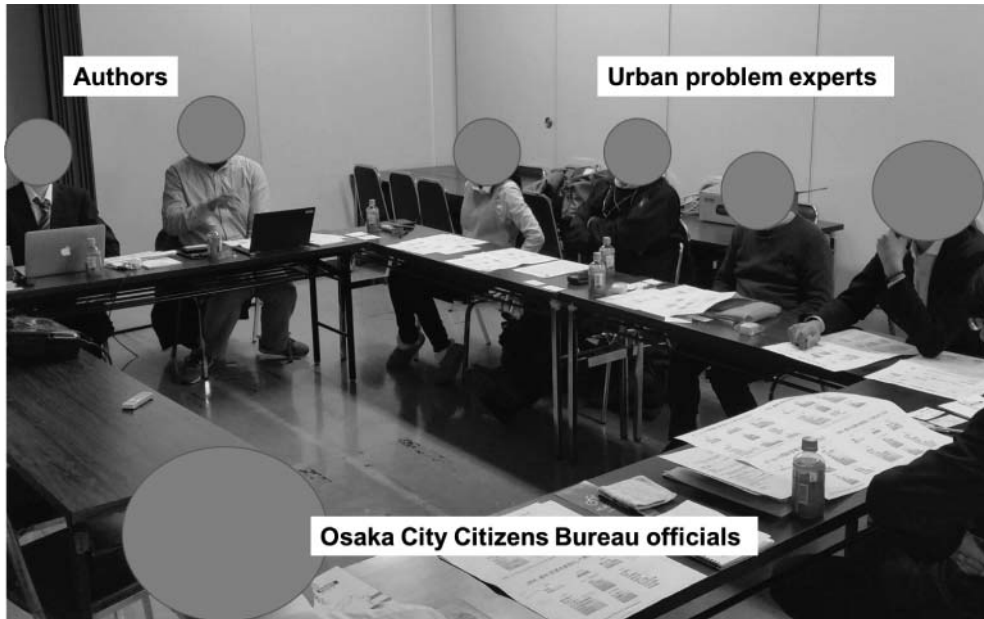
Figure 11. State of the evaluation of urban problem causalities.

We then evaluated the accuracy of the vicious cycles based on these results. In this paper, we defined the vicious cycles as consisting of the causal relations from Options 1 or 2. As a result, 196 of the detected vicious cycles related to homelessness and crime were correctly extracted. For example, "Poverty $\xrightarrow{\text{affects}}$ Poverty business $\xrightarrow{\text{affects}}$ Day labor $\xrightarrow{\text{affects}}$ Homelessness $\xrightarrow{\text{affects}}$ Disease" was detected as a vicious cycle that could occur. The temporary staffing business of day labor is one of the poverty businesses seeking to exploit the weakness of people already in difficulty [27]. Since day laborers cannot earn stable income, we can consider that they might become homelessness. Then, homeless may affect Disease, which may affect Poverty, again. Thus, we can consider that the increasing hospitalization expenses might lead to the poverty. However, a long-term survey is needed to determine if these vicious cycles are observed in the real world.

5.5 Evaluation of Detecting Root Problems

Consequently, we found that, for example, illegally parked bicycles can affect traffic accidents and littering; they were elements of vicious cycles as follows: "Traffic accident $\xrightarrow{\text{affects}}$ Traffic jam $\xrightarrow{\text{affects}}$ Stress", and "Littering $\xrightarrow{\text{affects}}$ Deteriorated security $\xrightarrow{\text{affects}}$ Graffiti". This problem has been actually identified as a factor of traffic accidents, safety security, and many other urban problems by several city bureaus in Japan and maybe in Asian countries. Since the illegally parked bicycle problem is one of the root problems, the solution to this problem might make a large positive impact on the city. Furthermore, we searched for budget information related to root problems that lead to multiple vicious cycles. As an example, we found that the truancy problem could lead to the vicious cycles "Poverty $\xrightarrow{\text{affects}}$

Homelessness $\xrightarrow{\text{affects}}$ Poverty business $\xrightarrow{\text{affects}}$ Day labor" (which consisted of only Options 1 and 2) and "Deterioration of security $\xrightarrow{\text{affects}}$ Graffiti $\xrightarrow{\text{affects}}$ Thief." Because truant children might not be able to find steady jobs, their truancy might lead to poverty in the future. Truancy also increases bad behavior and leads to the deterioration of security. Graffiti gives the public the sense that the local government is not functioning, which can lead to crimes such as theft. This phenomenon is well known as the broken window theory [25]. In fact, the experts in our study agreed that truancy and a lack of educational opportunities are root problems. However, according to the Osaka city manager's budget data, the budget for solving the truancy problem in Abeno ward was only 15,000 JPY. We obtained these results using a SPARQL query (Figure 12). The budget for solving the truancy problem was 15,083 JPY on average in the other wards. Therefore, increasing the budgets for these services could reduce the risk of children entering these vicious cycles.

```
PREFIX upr: http://www.ohsuga.lab.uec.ac.jp/urbanproblem/resource/>
PREFIX upq: http://www.ohsuga.lab.uec.ac.jp/urbanproblem/qb/>
PREFIX upv: <http://www.ohsuga.lab.uec.ac.jp/urbanproblem/vocabulary#>
PREFIX dcterms: <http://purl.org/dc/terms/>
SELECT * WHERE {
        ?project        dcterms:subject        upr:Truancy ;
                        upq:ward        ?ward ;
                        upq:generalRevenue   ?budget .
        ?ward           skos:broader   upq:大阪市 (Osaka city) .
}
```

Figure 12. SPARQL query for searching budgets.

Finally, we received agreements at the discussion, such as these from the experts and the Osaka City Citizens Bureau "This KG is useful when we recognize the overview of the urban problem, as the urban-problem experts sometimes have a certain mind-set" and "This KG is useful as a tool for improving discussions."

6. CONCLUSION

In this paper, we first described an ontology for urban-problem causality and for examining budgets and building a KG based on the ontology. The designed ontology enabled a search for the factor words and affecting words of urban problems. Then, to understand the structure of socially intertwined urban problems, we detected vicious cycles using SPARQL and inference rules. Afterward, we evaluated the results with the help of six experts on urban problems. Furthermore, to understand which problems should be resolved first, we proposed SPARQL patterns for detecting root problems and discussed the results of the root problem detection using budget information.

chinaXiv:202211.00405v1

In this study, we constructed urban-problem causality based on government documents, sociology articles, and social opinions from the Web. In the future, we will consider adding the probabilities of causal relations as numerical values.

ACKNOWLEDGEMENTS

AUTHOR CONTRIBUTIONS

S. Egami (s-egami@aist.go.jp), T. Kawamura (takahiro.kawamura@naro.go.jp), K. Kozaki (kozaki@osakac. ac.jp), and A. Ohsuga (ohsuga@is.uec.ac.jp) proposed the research problems, performed the research, designed the research framework, collected and analyzed the data and wrote and revised the manuscript.

REFERENCES

[1] Etcheverry, L., Vaisman, A., Zimányi, E.: Modeling and querying data warehouses on the semantic Web using QB4OLAP. In: The 16th International Conference on Data Warehousing and Knowledge Discovery (DaWaK), pp. 45–56 (2014)

[2] Szekely, P., et al.: Building and using a knowledge graph to combat human trafficking. In: The 14th International Semantic Web Conference (ISWC), pp. 205–221 (2015)

[3] Egami, S., Kawamura, T., Ohsuga, A.: Building urban LOD for solving illegally parked bicycles in Tokyo. In: The 15th International Semantic Web Conference (ISWC), pp. 291–307 (2016)

[4] Egami, S., et al.: Construction of linked urban problem data with causal relations using crowdsourcing. In: The 6th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI), pp. 814–819 (2017)

[5] Egami, S., et al.: Linked urban open data including social problems' causality and their cost. In: The 7th Joint International Semantic Technology Conference (JIST), pp. 334–349 (2017)

[6] Shiramatsu, S., et al.: Towards continuous collaboration on civic tech projects: Use cases of a goal sharing system based on linked open data. In: The 7th IFIP International Conference on Electronic Participation (ePart), pp. 81–92 (2015)

[7] Santos, H., et al.: From data to city indicators: A knowledge graph for supporting automatic generation of dashboards. In: The 14th Extended Semantic Web Conference (ESWC), pp. 94–108 (2017)

[8] Pileggi, S.F., Hunter, J.: An ontological approach to dynamic fine-grained urban indicators. In: The 17th International Conference on Computational Science (ICCS), pp. 2059–2068 (2017)

[9] Höffner, K., Martin, M., Lehmann, J.: Linkedspending: Openspending becomes linked open data. Semantic Web Journal 7(1), 95–104 (2016)

[10] Demartini, G., Difallah, D.E., Cudré-Mauroux, P.: Large-scale linked data integration using probabilistic reasoning and crowdsourcing. The International Journal on Very Large Data Bases 22(5), 665–687 (2013)

[11] Celino, I., et al.: Linking smart cities datasets with human computation—the case of urbanmatch. In: The 11th International Semantic Web Conference (ISWC), pp. 34–49 (2011)

[12] Ahn, L.V.: Games with a purpose. IEEE Computer 39(6), 92–94 (2006)

[13] Nguyen, T.M., et al.: Self-supervised capturing of users' activities from weblogs. International Journal of Intelligent Information and Database Systems 6(1), 61–76 (2012)

[14] Augenstein, I., Padó, S., Rudolph, S.: Lodifier: Generating linked data from unstructured text. In: The 9th Extended Semantic Web Conference (ESWC), pp. 210–224 (2012)

[15] Milne, D., Witten, I.H.: Learning to link with Wikipedia. In: The 17th ACM Conference on Information and Knowledge Management (CIKM), pp. 509–518 (2008)

[16] Strauss, B., et al.: Results of the WNUT16 named entity recognition shared task. In: The 2nd Workshop on Noisy User-generated Text (WNUT), pp. 138–144 (2016)

[17] Moitra, A., et al.: Semantic inference for pharmacokinetic drug-drug interactions. In: The 8th IEEE International Conference on Semantic Computing (ICSC), pp. 92–95 (2014)

[18] Herrero-Zazo, M., et al.: Dinto: Using OWL ontologies and SWRL rules to infer drug-drug interactions and their mechanisms. Journal of Chemical Information and Modeling 55(8), 1698–1707 (2015)

[19] Kudo, T., Matsumoto, Y.: Japanese dependency analyisis using cascaded chunking. In: The 6th Conference on Natural Language Learning (CoNLL), pp. 1–7 (2002)

[20] Winkler, W.: The state record linkage and current research problems. Technical report, Statistics of Income Division, Internal Revenue Service Publication (1999)

[21] Ponzetto, S.P., Strube, M.: Deriving a large scale taxonomy from Wikipedia. In: The 22nd AAAI Conference on Artificial Intelligence (AAAI), pp. 1440–1445 (2007)

[22] Kontokostas, D., et al.: Test-driven evaluation of linked data quality. In: The 23rd International Conference on World Wide Web (WWW), pp. 747–758 (2014)

[23] Fleiss, J.L., Cohen, J.: The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. Educational and Psychological Measurement 33(3), 613–619 (1973)

[24] Viera, A.J., Garrett, J.M.: Understanding interobserver agreement: The kappa statistic. Family Medicine 37(5), 360–363 (2005)

[25] Wilson, J.Q., George, L.K.: Broken windows. In: Dunham, R.G., Albert, G.P. (eds.) Critical Issues in Policing: Contemporary Readings, pp. 369–381 (1989)

[26] Egami, S., et al.: Urban problem LOD for understanding the problem structure and detecting vicious cycles. In: The 12th IEEE International Conference on Semantic Computing (ICSC), pp. 186–193 (2018)

[27] Sekine, , Y.: The rise of poverty in Japan: The emergence of the working poor. Japan Labor Review 5(4), 49–66 (2008)

AUTHOR BIOGRAPHY

Shusaku Egami received his Ph.D. degree in Computer Science from the University of Electro-Communications, Tokyo, Japan, in 2019. He is currently a researcher at the National Institute of Advanced Industrial Science and Technology, Japan. He is also a part-time lecturer at Hosei University, Tokyo, Japan, and a collaborative researcher at the University of Electro-Communications, Tokyo, Japan. His research interests include the semantic Web, knowledge graph, and ontology.
ORCID: 0000-0002-3821-6507

Takahiro Kawamura is an Administrator of Data Management, National Agriculture and Food Research Organization, a member of Evaluation Committee, Riken BioResource Research Center, a Visiting Researcher at Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology, and a Part-time Lecturer in the Graduate School of Engineering, Osaka University, Japan. He received M.S. and Ph.D. degrees in electrical engineering from Waseda University. From 2001 to 2002 he was a Visiting Scholar at Robotics Institute, Carnegie Mellon University. His current research interests include knowledge graph construction and its analysis combining semantic technology and machine learning. He was a board member of the Japanese Society for Artificial Intelligence (JSAI) and served as a chair of SIG on Semantic Web and Ontology in JSAI. He was also a former W3C Advisory Committee Representative and a member in the Program/Organizing Committees of numerous conferences and workshops including the International Semantic Web conference.
ORCID: 0000-0002-2765-6232

Kouji Kozaki is the professor of Osaka Electro-Communication University (OECU), Osaka, Japan. His field of expertise is Computer Science, in particular, Ontology Engineering and Semantic Technologies. His research interest includes ontology building, fundamental theory of ontological engineering (especially for role and identity), ontology development system, and ontology-based application. A major theme of his research is a development Hozo (http://www.hozo.jp/), which is an environment for building/using ontologies, and ontology building in several domains such as clinical medicine, bioinformatics, and environmental engineering. He was also a co-Program Committee chair of JIST (Joint International Semantic Technology Conference) 2015 and a co-General Chair of JIST2018.

ORCID: 0000-0003-4578-4980

Akihiko Ohsuga received his Ph.D. degree in Computer Science from Waseda University in 1995. From 1981 to 2007 he was with Toshiba Corporation. He joined the University of Electro-Communications in 2007. He is currently a professor in the Graduate School of Informatics and Engineering. He is also a visiting professor at National Institute of Informatics. His research interests include agent technologies, Web intelligence, and software engineering. He is a member of IEEE Computer Society (IEEE CS), Information Processing Society of Japan (IPSJ), Institute of Electronics, Information and Communication Engineers (IEICE), Japanese Society for Artificial Intelligence (JSAI), Japan Society for Software Science and Technology (JSSST), and Institute of Electrical Engineers of Japan (IEEJ). He has been a fellow of IPSJ since 2017. He served as a chair of IEEE CS Japan Chapter, a member of JSAI board of directors, a member of JSSST board of directors, and a member of JSSST councilor. He received IPSJ Best Paper Awards in 1987 and 2017.

ORCID: 0000-0001-6717-7028